

Remarks

This REPLY is in response to the Office Action mailed May 12, 2008 and an interview with Examiner Jason Mitchell on September 8, 2008. Applicant acknowledges the courtesy of an interview with the Examiner, during the course of which interview several amendments to the claims were discussed, the substance of which amendments are set forth fully herein.

I. Summary of Examiner's Rejections

In the Office Action mailed May 12, 2008, Claims 1-4, 6-11, 13-18, and 20-24 were rejected under 35 U.S.C. 112 as being indefinite. Claims 1-4, 6-11, 12-13, and 20-24 were also rejected under 35 U.S.C. 103(a) as being unpatentable over McNeely et al. (U.S. Patent Number 7,117,411 hereafter McNeely) in view of Dubovsky (U.S. Patent Publication Number 2003/0055836).

II. Summary of Applicant's Amendment

The present Reply cancels Claims 1, 4, 8, 11, 15, and 18, leaving for the Examiner's present consideration Claims 1-4, 6-11, 13-18, and 20-24. Reconsideration of the Application, as amended, is respectfully requested.

III. Claim Rejections under 35 U.S.C. 112

In the Office Action mailed May 12, 2008, Claims 1-4, 6-11, 13-18, and 20-24 were rejected under 35 U.S.C. 112 as being indefinite. Accordingly, Claims 1-4, 6-11, 13-18, and 20-24 have been amended as shown above. Applicant respectfully submits that Claims 1-4, 6-11, 13-18, and 20-24 now conform to the requirements of 35 U.S.C. 112 and reconsideration thereof is respectfully requested.

IV. Claim Rejections under 35 U.S.C. 103(a)

In the Office Action mailed May 12, 2008, Claims 1-4, 6-11, 12-13, and 20-24 were also rejected under 35 U.S.C. 103(a) as being unpatentable over McNeely (U.S. Patent Number 7,117,411) in view of Dubovsky (U.S. Patent Publication Number 2003/0055836).

Claim 1

Claim 1 has been amended to more clearly define the embodiment therein. As amended, Claim 1 defines:

1. *(Currently Amended) A system that provides a generic user interface testing framework, and allows a user to test and debug graphical user interfaces for software applications under development, comprising:*

a computer including a computer readable medium, and a processor operating thereon;

a software application source code, stored on the computer readable medium, wherein the software application source code defines a software application under development, including a graphical user interface as part of the software application, and wherein the software application source code executes on the computer to display its graphical user interface;

a plurality of different software test tools, wherein each software test tool is associated with a different tool-specific scripting language, that can be invoked by a user to perform testing operations on the graphical user interface that is displayed while the software application is running, and wherein each of the plurality of different software test tools use only their associated tool-specific scripting language to test graphical user interfaces associated with a plurality of different software applications;

a test case input file stored on the computer readable medium, that contains a plurality of generic interface commands that are abstractions independent of any of the tool-specific scripting languages, wherein the test case input file can be edited and reused as necessary by the user to specify different generic interface commands for testing against a software application's graphical user interface in any of the different software test tools; and

an interpretive engine that executes on the computer, and that includes a plurality of dynamically loaded libraries corresponding to the plurality of different software test tools, and including a library for each of the plurality of different software test tools, wherein the interpretive engine receives the generic interface commands defined in the test case input file, loads required libraries associated with the software test tool the user is currently using, maps the generic interface commands to the software test tool's associated tool-specific scripting language, uses the software test tool to perform the testing operations on the software application's graphical user interface using the associated tool-specific scripting language, and reports to the user the success or failure of the testing operations.

Claim 1, as amended, defines a plurality of different software test tools, wherein each software test tool is associated with a different tool-specific scripting language. Claim 1, as amended, further defines each of the plurality of different software test tools use only their associated tool-specific scripting language to test graphical user interfaces associated with a plurality of different software applications. Additionally, Claim 1, as amended, defines that the interpretive engine receives the generic interface commands defined in the test case input file, loads required libraries associated with the software test tool the user is currently using, maps the generic interface commands to the software test tool's associated tool-specific scripting language, uses the software test tool to perform the testing operations on the software application's graphical user interface using the associated tool-specific scripting language, and reports to the user the success or failure of the testing operations.

McNeely discloses a generalized test environment that allows complete automation of

test cases. The test cases are independent of the number or types of devices under test. As such, a test operator responsible for writing a test script need not know the device-specific commands because test environment device packages map the device-specific commands to a common scripting language. Therefore, the operator need only be familiar with a common script language rather than device-specific test commands for multiple devices. (Column 3, lines 55-64).

Dubovsky discloses methods for generating data structures for use with an environment based data driven test engine for computer programs which have a graphical user interface (GUI). (Abstract).

As described above, McNeely appears to disclose a single test tool that understands a common script language, specific to the test tool. The test tool enables a user to test different devices without having specific knowledge of each devices' testing interface. The tool appears to enable this functionality using a common script language that is specific to the McNeely test environment. This way, a user does not need to be familiar with each device-specific language for each device he is testing. Dubovsky appears to disclose methods for testing programs which have GUIs. In contrast, Claim 1 has been amended to more clearly define a plurality of different software test tools, wherein each of the plurality of different software test tools have their own test environment and understand their own tool-specific scripting language.

Applicant respectfully submits that the combination of McNeely in view of Dubovsky appears to disclose a single test tool with a tool-specific scripting language. Claim 1, however, defines a plurality of different software test tools and each software test tool is associated with a different tool-specific scripting language. Claim 1 also defines generic interface commands that are abstractions of any of the tool-specific scripting language. Thus, a user does not need knowledge of either the application-specific environment he is testing or the test-tool environment used to conduct the test. Instead the user only needs to know the generic interface commands. Further, this means that the user is not locked into a single test tool; the same, previously written scripts can be used if the user stops using a first test tool that is only compatible with a first test-tool specific scripting language and begins using a second test tool that is only compatible with a second test-tool specific scripting language. Applicant respectfully submits that the cited references, alone or in combination, do not disclose or render obvious the embodiment of Claim 1, as amended.

In view of the above comments, Applicant respectfully submits that Claim 1, as currently amended, is neither anticipated by nor obvious in view of the cited references, and reconsideration thereof is respectfully requested.

Claims 8 and 15

The comments provided above with respect to Claim 1 are hereby incorporated by reference. Claims 8 and 15 have been similarly amended to more clearly define the embodiments therein. For similar reasons as provided above with respect to Claim 1, Applicant respectfully submits that Claims 8 and 15, as amended, are likewise neither anticipated by, nor obvious in view of the cited references, and reconsideration thereof is respectfully requested.

Claims 4, 11, and 18

Claims 4, 11, and 18 have been amended to more clearly define the embodiments therein. As amended, Claims 4, 11, and 18 define a rules-based wizard that guides the user to edit or create the test case input file by choosing the testing operations to be included in the test case input file wherein the rules-based wizard maps the testing operations to generic interface commands. Applicant respectfully submits that the cited references, alone or in combination, do not disclose or render obvious these features.

Claims 2-3, 6-7, 9-10, 13-14, 16-17, and 20-24

Claims 2-3, 6-7, 9-10, 13-14, 16-17, and 20-24 depend from and include all of the features of Claims 1, 8, or 15. Claims 2-3, 6-7, 9-10, 13-14, 16-17, and 20-24 have not been addressed separately but it is respectfully submitted that these claims are allowable as depending from an allowable independent claim, and further in view of the comments provided above. Reconsideration thereof is respectfully requested.

V. Conclusion

In light of the above, it is respectfully submitted that all of the claims now pending in the subject patent application should be allowable, and reconsideration thereof is respectfully requested. The Examiner is respectfully requested to telephone the undersigned if he can assist in any way in expediting issuance of a patent.

Enclosed is a PETITION FOR EXTENSION OF TIME UNDER 37 C.F.R. § 1.136 for extending the time to respond up to and including today, September 12, 2008.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: September 12, 2008

By: /Nathan L. Feld/
Nathan L. Feld
Reg. No. 59,725

FLIESLER MEYER LLP
650 California Street, 14th Floor
San Francisco, California 94108
Telephone: (415) 362-3800
Customer No. 80548